**“ZOMATO - PREDICTING RESTAURANT’S POPULARITY”**

**SUMMER TRAINING PROJECT REPORT**

## By

**Mohammad Noman CDAC, NOIDA**

***Under the guidance of***

## Mr. Rahul Mishra

(School of IT)



# CENTRE FOR DEVELOPMENT OF ADVANCE COMPUTING, NOIDA

# CANDIDATE’S DECLARATION

I hereby certify that the project work entitled **“ZOMATO - PREDICTING RESTAURANT’S POPULARITY”** is my own bona fide work carried out by me under the supervision of Mr. Rahul Mishra, CDAC, Noida during the period of 1st June 2019 to 1st August 2019. This work has not been submitted elsewhere for the award of a degree/diploma/certificate.

Mohammad Noman

MCA,

2nd Semester

CDAC, Noida

This is to certify that the above-mentioned statement in the candidate’s declaration is correct to the best of my knowledge.

Date:

### Mr. Rahul Mishra

Name and Signature of Mentor

# ACKNOWLEDGEMENT

I wish to express our deep gratitude to our guide **Mr. Rahul Mishra**, for all advice, encouragement and constant support he has given us throughout our project work. He constantly made time for us in his hectic schedule. This work would not have been possible without his support and valuable suggestions.

I would most of all like to thank all my **team members** for their varied suggestions sans whom the project wouldn’t have been possible at all. I would like to thank my friends for their help and constructive criticism during our project work. Finally, we have no words to express our sincere gratitude to our **parents** who have shown us this world and for every support they have given us.

# TABLE OF CONTENTS

## S. No. Contents Page No.

[Candidate’s declaration ii](#_TOC_250007)

[Acknowledgment iii](#_TOC_250006)

[Table of Contents v](#_TOC_250005)

List of Figures vii

1. [Introduction 5](#_TOC_250004)
   1. [About Zomato Dataset 5](#_TOC_250003)
   2. [Attributes 6](#_TOC_250002)
   3. [Objective 6](#_TOC_250001)
2. [System analysis 7](#_TOC_250000)
   1. System requirements
   2. Hardware Requirements
   3. Software Requirements
3. **Models Implemented 8**
   1. Naïve Bayes
   2. Logistic Regression
   3. K-Nearest Neighbor
   4. Decision tree
   5. Random Forest
4. **Implementation 12**
5. **Result 17**
6. **Conclusion 23**

# INTRODUCTION

### About Zomato Dataset

### The basic idea of analyzing the Zomato dataset is to get a fair idea about the factors affecting the aggregate rating of each restaurant, the establishment of different types of the restaurant at different places, Bengaluru being one such city has more than 12,000 restaurants with restaurants serving dishes from all over the world. With each day new restaurants opening the industry hasn’t been saturated yet and the demand is increasing day by day. In spite of increasing demand, it however, has become difficult for new restaurants to compete with established restaurants. Most of them serving the same food. Bengaluru being an IT capital of India. Most of the people here are dependent mainly on the restaurant food as they don't have time to cook for themselves. With such an overwhelming demand for restaurants, it has therefore become important to study the demography of a location. What kind of food is more popular in a locality. Do the entire locality love vegetarian food. If yes then is that locality populated by a particular set of people for eg. Jain, Marwaris, Gujaratis who are mostly vegetarian. This kind of analysis can be done using the data, by studying different factors.

### Dataset Attributes

Columns

* URL - contains the URL of the restaurant in the zomato website
* address - contains the address of the restaurant in Bengaluru
* name - contains the name of the restaurant
* online\_order - whether online ordering is available in the restaurant or not
* book\_table - table book option available or not
* rate - contains the overall rating of the restaurant out of 5
* votes - contains the total number of rating for the restaurant as of the above-mentioned date
* phone - contains the phone number of the restaurant
* location - contains the neighborhood in which the restaurant is located
* rest\_type - restaurant type
* dish\_liked - dishes people liked in the restaurant
* cuisines - food styles, separated by a comma
* approx\_cost(for two people) - contains the approximate cost for a meal for two people
* reviews\_list - list of tuples containing reviews for the restaurant, each tuple consists of two values, rating and review by the customer
* menu\_item - contains a list of menus available in the restaurant
* listed\_in(type) - type of meal
* listed\_in(city) - contains the neighborhood in which the restaurant is listed

### Objective

To make a prediction about the expected rating of a restaurant to be opened with given cuisines and menu.

# SYSTEM ANALYSIS

### System requirements

Client – Browser with internet connectivity

### Hardware Requirements

Minimum 4GB RAM

### Software Requirements

Anaconda3 for Jupyter Notebook

Python 3

# MODELS IMPLEMENTED

**3.1 NAÏVE BAYES**

Naive Bayes classifiers have been popular for text classification, and are a traditional solution for problems such as spam detection.

In the context of text classification, where features may be word counts, features may follow a **multinomial distribution**. In other cases, where features are continuous, they may follow a **Gaussian distribution**.

Scikit learn (python library) will help here to build a Naive Bayes model in Python. There are three types of Naive Bayes model under sci-kit learn library:

* [**Gaussian:**](http://scikit-learn.org/stable/modules/naive_bayes.html)It is used in classification and it assumes that features follow a normal distribution.
* [**Multinomial**](http://scikit-learn.org/stable/modules/naive_bayes.html)**:**It is used for discrete counts. For example, let’s say,  we have a text classification problem. Here we can consider Bernoulli trials which is one step further and instead of “word occurring in the document”, we have “count how often word occurs in the document”, you can think of it as “number of times outcome number x\_i is observed over the n trials”.
* [**Bernoulli**](http://scikit-learn.org/stable/modules/naive_bayes.html)**:**The binomial model is useful if your feature vectors are binary (i.e. zeros and ones). One application would be text classification with ‘bag of words’ model where the 1s & 0s are “word occurs in the document” and “word does not occur in the document” respectively.

FUNCTIONS USED

* MultinomialNB()
* BernoulliNB()
* GaussianNB()

**3.2 LOGISTIC REGRESSION**

 Logistic Regression is used when the dependent variable(target) is categorical.

For example,

* To predict whether an email is a spam (1) or (0)
* Whether the tumor is malignant (1) or not (0)

FUNCTION USED:

LogisticRegression()

**3.3 K- Nearest Neighbors (KNN)**

KNN (K — Nearest Neighbors) is one of many (supervised learning) algorithms used in data mining and machine learning, it’s a classifier algorithm where the learning is based “how similar” is a data (a vector) from other.

FUNCTION USED:

KNeighborsClassifier()

**3.4 Decision Tree**

*Decision Tree Classifier repetitively divides the working area(plot) into subpart by identifying lines.*

Impurity is when we have a traces of one class division into other. This can arise due to the following reason

1. We run out of available features to divide the class upon.
2. We tolerate some percentage of impurity (we stop further division) for faster performance. (There is always trade-off between accuracy and performance).

For example in second case we may stop our division when we have x number of fewer number of elements left. This is also known as ***Gini impurity.***

Entropy is degree of randomness of elements or in other words it is a **measure of impurity.**

FUNCTION USED:

* DecisionTreeClassifier()
* DecisionTreeClassifier(criterion = 'entropy')

**3.5 RANDOM FOREST**

* It is unexcelled in accuracy among current algorithms.
* It runs efficiently on large databases.
* It can handle thousands of input variables without variable deletion.
* It gives estimates of what variables are important in the classification.
* It generates an internal unbiased estimate of the generalization error as the forest building progresses.
* It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
* It has methods for balancing error in class population unbalanced data sets.
* Generated forests can be saved for future use on other data.
* Prototypes are computed that give information about the relation between the variables and the classification.
* It computes proximities between pairs of cases that can be used in clustering, locating outliers, or (by scaling) give interesting views of the data.
* The capabilities of the above can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection.
* It offers an experimental method for detecting variable interactions.

FUNCTION USED:

RandomForestRegressor()

1. **IMPLEMENTATION**

**4.1 DATA PREPROCESSING**

**LIBRARIES USED**

* Pandas
* Numpy
* First, we checked whether we have null values in our dataset or not. We did that using the **isnull()** method.

**zomato.isnull()**

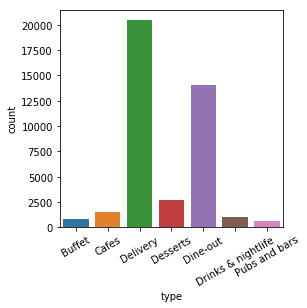
* Then replacing the rows or columns that contain null values with a median.

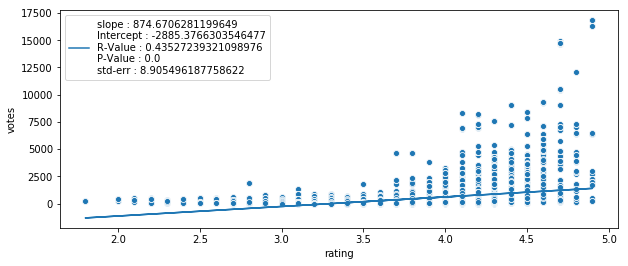
*zomato['approx\_cost(for two people)'].replace(np.NaN,’’) zomato['approx\_cost(for two people)'].median())*

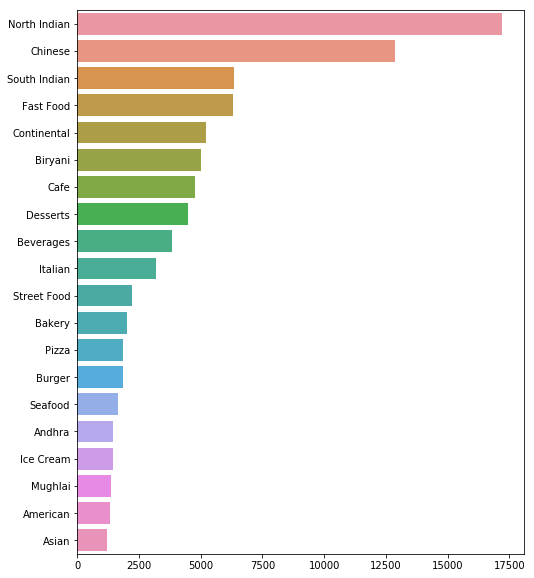
* Next, the rows which do not affect our prediction were dropped.

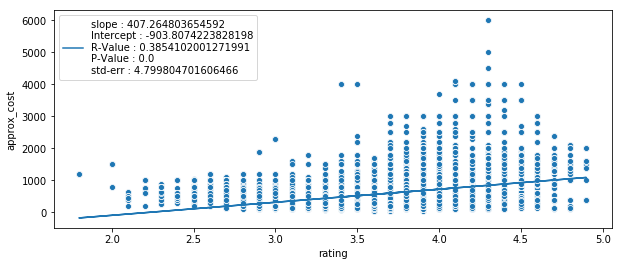
*zomato.drop(['url', 'cost\_for\_2', 'index','address','name','online\_order','book\_table','votes','phone','reviews\_list','menu\_item','listed\_in(type)','listed\_in(city)'], inplace = True,axis=1)*

**These steps gave us the final preprocessed data which is ready for applying models.**

**4.2 DATA EXPLORATION AND VISUALIZATION**

****

****

****

**4.3 TRAINING AND FITTING**

**LIBRARIES USED**

* Pandas
* Numpy
* Sklearn
* ***Training Dataset****: The sample of data used to fit the model.*
* The actual dataset that we use to train the model. The model sees and learns from this data.
* We divided our dataset into the 80-20 ratio.

*train\_test\_split(x,y , test\_size=0.2)*

* **Fitting Dataset**

*v.fit\_transform(X\_train)*

*v.transform(X\_test)*

*rf.fit(x\_train\_tf, y\_train.astype(float))*

**4.4 PREDICTING AND FINDING ACCURACY**

*y\_pred\_rf= rf.predict(x\_test\_tf)*

*print('R2 score:', r2\_score(y\_test, y\_pred\_rf)\*100, '%')*

1. **RESULT**

* Least accurate: NAÏVE BAYES- 38%
* Most accurate: RANDOM FOREST VECTOR- 92%

**Random Forests, as compared to Naive Bayes:**

1. Offers consistent and marked improvements in accuracy

○ Particularly true for multiple class classification tasks

2. Requires far more processing time

○ At least an order of magnitude

○ Computational burden increases with no. of features

**6. CONCLUSION**

The prediction on Zomato Dataset has been successfully implemented and analyzed.

**7. REFERENCES**

1. [www.medium.com](http://www.medium.com)

2. [www.kaggle.com](http://www.kaggle.com)

3. [www.queirozf.com](http://www.queirozf.com)

4. [www.geeksforgeeks.com](http://www.geeksforgeeks.com)

5. [www.edureka.com](http://www.edureka.com)

6. [www.pythonprogramming.net](http://www.pythonprogramming.net)

7. [www.coursera.com](http://www.coursera.com)